

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of:

Jae-seong Shim et al.

Application No.: TO BE ASSIGNED

Group Art Unit: TO BE ASSIGNED

Filed: September 23, 2003

Examiner:

For: DATA MODULATING METHOD AND APPARATUS, DATA DEMODULATING METHOD
AND APPARATUS, AND CODE ARRANGING METHOD

**SUBMISSION OF CERTIFIED COPY OF PRIOR FOREIGN
APPLICATION IN ACCORDANCE
WITH THE REQUIREMENTS OF 37 C.F.R. § 1.55**

Commissioner for Patents
PO Box 1450
Alexandria, VA 22313-1450

Sir:

In accordance with the provisions of 37 C.F.R. § 1.55, the applicant(s) submit(s)
herewith a certified copy of the following foreign application:

Korean Patent Application No(s).

2002-58809 filed September 27, 2002 and

2002-63534 filed October 17, 2002

It is respectfully requested that the applicant(s) be given the benefit of the foreign filing
date(s) as evidenced by the certified papers attached hereto, in accordance with the
requirements of 35 U.S.C. § 119.

Respectfully submitted,

STAAS & HALSEY LLP

Date: September 23, 2003

By: 

Michael D. Stein

Registration No. 37,240

1201 New York Ave, N.W., Suite 700
Washington, D.C. 20005
Telephone: (202) 434-1500
Facsimile: (202) 434-1501

대한민국 특허청

KOREAN INTELLECTUAL PROPERTY OFFICE

별첨 사본은 아래 출원의 원본과 동일함을 증명함.

This is to certify that the following application annexed hereto
is a true copy from the records of the Korean Intellectual
Property Office.

출원번호 : 10-2002-0058809
Application Number PATENT-2002-0058809

출원년월일 : 2002년 09월 27일
Date of Application SEP 27, 2002

출원인 : 삼성전자 주식회사
Applicant(s) SAMSUNG ELECTRONICS CO., LTD.

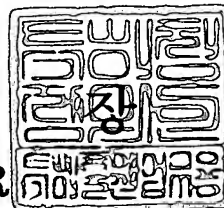
SH



2002 년 11 월 28 일

특 허 청

COMMISSIONER



【서지사항】

【서류명】	특허출원서
【권리구분】	특허
【수신처】	특허청장
【참조번호】	0014
【제출일자】	2002.09.27
【국제특허분류】	H04N
【발명의 명칭】	데이터 변조 방법과 장치 및 코드 배치 방법
【발명의 영문명칭】	Data modulation method and apparatus the same, and code alignment method
【출원인】	
【명칭】	삼성전자 주식회사
【출원인코드】	1-1998-104271-3
【대리인】	
【성명】	이영필
【대리인코드】	9-1998-000334-6
【포괄위임등록번호】	1999-009556-9
【대리인】	
【성명】	이해영
【대리인코드】	9-1999-000227-4
【포괄위임등록번호】	2000-002816-9
【발명자】	
【성명의 국문표기】	심재성
【성명의 영문표기】	SHIM, Jae Seong
【주민등록번호】	641223-1058515
【우편번호】	143-191
【주소】	서울특별시 광진구 자양1동 610-35호
【국적】	KR
【발명자】	
【성명의 국문표기】	김진한
【성명의 영문표기】	KIM, Jin Han
【주민등록번호】	740217-1691317
【우편번호】	441-390

【주소】 경기도 수원시 권선구 권선동 1321번지 대림아파트 221동 1004호
【국적】 KR
【발명자】
【성명의 국문표기】 정규해
【성명의 영문표기】 JUNG,Kiu Hae
【주민등록번호】 740605-1466315
【우편번호】 442-374
【주소】 경기도 수원시 팔달구 매탄4동 214-26번지 201호
【국적】 KR
【취지】 특허법 제42조의 규정에 의하여 위와 같이 출원합니다. 대리인
 필 (인) 대리인 이영
 이해영 (인)
【수수료】
【기본출원료】 20 면 29,000 원
【가산출원료】 20 면 20,000 원
【우선권주장료】 0 건 0 원
【심사청구료】 0 항 0 원
【합계】 49,000 원
【첨부서류】 1. 요약서·명세서(도면)_1통

【요약서】**【요약】**

본 발명에는 데이터 변조 방법과 장치 및 코드 배치 방법이 개시되어 있다.

본 발명은 m 비트의 소스데이터를 최소 구속장 d 와 최대 구속장 k 로 제한하면서 n 비트($n \geq m$)의 코드워드로 변환하는 데이터 변조 방법에 있어서, 입력 데이터열을 일정 길이로 분할하고, 분할된 입력 데이터열을 다중화 정보를 이용하여 소정의 다중화 방식에 의해 다중화해서 다중화된 데이터열을 제공하는 단계와 다중화된 데이터열에 대해 별도의 부가 비트가 부가된 DC 제어 변환표를 사용하지 않는 weak DC-free RLL(Run Length Limited) 변조를 수행하고, 다중화되고 RLL 변조된 코드열 중에서 가장 DC 성분이 작은 코드열을 제공하는 단계를 포함한다. 따라서, 본 발명은 불충분한 DC 억압 변조 코드에 대해 멀티모드 코딩 방식을 결합하여 DC 억압 능력을 향상시킨 고효율의 변조 코드를 제공함으로써 기록밀도 측면에서 높은 효율성을 제공한다.

【대표도】

도 2

【명세서】**【발명의 명칭】**

데이터 변조 방법과 장치 및 코드 배치 방법{Data modulation method and apparatus the same, and code alignment method}

【도면의 간단한 설명】

- 도 1은 종래 코드들의 파워 스펙트럼 밀도(PSD) 곡선을 보인 도면,
- 도 2는 본 발명에 의한 데이터 변조 장치의 일 실시 예에 따른 블록도,
- 도 3은 주 변환 코드 그룹들의 코드워드 특성을 정리한 표,
- 도 4는 DC 제어용 보조 변환 코드 그룹들의 코드워드 특성을 정리한 표,
- 도 5는 엔드 제로수(EZ)에 따라 결정되는 다음 코드 그룹(ncg)을 정리한 표,
- 도 6은 코드워드 a와 b가 연결될 때 구속장의 조건을 설명하기 위한 도면,
- 도 7은 도 6을 통해 구속장의 조건을 만족하지 못하는 경우 코드 변환 전과 후에 따른 파라미터 INV의 변화를 보인 표,
- 도 8은 DC 제어가능한 코드워드 b1, b2로 인한 코드열의 분기 예,
- 도 9는 도 2에 도시된 동기 및 다중화 ID 삽입기에서 다중화 정보가 다중화 ID로의 변환의 예를 보인 변환표,
- 도 10a 내지 도 10e는 상술한 사항들을 고려하여 생성 및 배치한 주 변환 코드표,
- 도 11은 상술한 사항들을 고려하여 생성 및 배치한 DC 제어용 보조 변환 코드표,
- 도 12는 본 발명의 RLL(1,7) 코드의 PSD 곡선을 보인 도면,

도 13은 본 발명의 RLL(1,7) 코드와 기존의 코드와의 기록 밀도와 기록 효율을 비교한 표,

도 14는 본 발명의 RLL(2,10) 코드의 PSD 곡선을 보인 도면,

도 15는 본 발명의 RLL(2,10) 코드와 기존의 EFMP 코드와의 기록 밀도와 기록 효율을 비교한 표이다.

【발명의 상세한 설명】

【발명의 목적】

【발명이 속하는 기술분야 및 그 분야의 종래기술】

<16> 본 발명은 n비트의 소스 코드를 m비트의 채널 코드로 변조하는 방법과 장치 및 코드 배치 방법에 관한 것으로, 특히 높은 코드 효율과 DC 억압 능력이 뛰어난 변조 코드를 제공하는 변조 방법과 장치 및 코드 배치 방법에 관한 것이다.

<17> DC 억압 능력이 없는 변조 코드에 DC 억압 능력을 부여하기 위한 방법 중 멀티모드(multimode) 코딩 방식이 있다. 이는 입력 데이터열에 a비트의 부가 정보를 삽입하고, 이 부가 정보에 따라 2^a 가지의 다른 랜덤 데이터열을 만들고 그 2^a 가지의 랜덤 데이터열에 DC 억압 능력이 없는 변조를 수행하더라도 그 중에서 가장 DC 성분이 적은 변조된 데이터열을 선택함으로써 DC 억압 능력을 갖도록 하는 방법이다.

<18> 종래에는 미합중국 특허번호 6,225,921호의 "Device for encoding/decoding n-bit source words into corresponding m-bit channel words, and vice versa"에서 예시한 $d=1$, $k=7$, $m=2$, $n=3$ 인 코드의 코드율(R)은 약 2%의 리던던시가 포함되면

$R=49/75=0.6533$ 이며, 코드효율(Code Efficiency: $R/C(d,k)$)은 $R/C(d,k)=0.6533/0.6793=96.2\%$ 이다. 미합중국 특허번호 6,225,921호에서 사용된 변조 코드를 설명의 편의상 이하 "A-Code"라고 지칭한다.

<19> 미합중국 특허번호 6,281,815 호의 "Method of allocating RLL code having enhanced DC suppression capability, modulation method, demodulation method, and demodulation apparatus therefor"에서 예시한 $d=1$, $k=8$, $m=8$, $n=12$ 인 코드의 코드율(R)은 약 2%의 리턴던시가 포함되면 $R=32/49=0.6531$ 이며 코드효율($R/C(d,k)$)은 $R/C(d,k)=0.6531/0.6853=95.3\%$ 이다. 미합중국 특허번호 6,281,815호에서 사용된 변조 코드를 설명의 편의상 이하 "B-Code"라고 지칭한다. C 는 코드의 d 와 k 에 따른 Capacity를 의미한다.

<20> 또한, Kees A. Schouhamer Immink, "Codes for Mass Data Storage Systems," Shannon Foundation Publishers, 1999의 13장에 설명한 Guided Scrambling 방법을 사용하여 데이터 25바이트마다 리턴던시 4비트를 삽입하고 $d=1$, $k=7$ 변조를 수행하는 경우 코드율(R)은 $R=200/306=0.6536$ 이며, 코드효율($R/C(d,k)$)은 $R/C(d,k)=0.6536/0.6793=96.2\%$ 이다. 위 문헌에서 사용된 변조코드를 설명의 편의상 이하 "C-Code"라고 한다.

<21> 상기의 종래 기술 3가지의 변조 방법에 따른 코드 효율이 95.3%~96.2%로 유사하고 이들 코드의 DC 억압 능력을 나타내는 파워 스펙트럼 밀도(PSD) 곡선은 도 1에 도시된 바와 같다.

<22> 그러나, 상술한 문헌에서 예시한 멀티모드 코딩 방식도 충분한 DC 억압 능력

을 가지려면 데이터열을 랜덤 데이터로 만들기 위한 부가 정보의 빈도수를 그만큼 높여야 한다. 또한, 높은 코드 효율을 갖는 변조 기법을 개발하더라도 DC 억압 능력이 충분하지 않은 경우도 있다. 그 예로, 상술한 미합중국 특허번호 6,281,815호에 개시된 B-Code도 리던던시가 없으면 DC 억압은 가능하나 별도의 부가 비트 없이는 만족할 만한 DC 억압 성능을 갖지 못하고 있다. 이하, 리던던시가 없으면 DC 억압은 가능하나 별도의 부가 비트가 없어 억압 성능은 떨어지는 코드를 불충분한 DC 억압 변조 코드(dc-free modulation code)라고 지칭한다.

【발명이 이루고자 하는 기술적 과제】

<23> 따라서, 본 발명의 목적은 DC 억압 성능은 3가지의 종래 기술과 같이 유지하면서 불충분한 DC 억압 변조 코드와 멀티모드 코딩 방식을 결합하여 고효율(high-efficient) 이면서도 DC 억압 능력이 뛰어난 변조 코드를 제공하는 데이터 변조 방법과 장치를 제공하는 데 있다.

<24> 본 발명의 다른 목적은 불충분한 DC 억압 변조 코드를 배치함에 있어서, 구속장 제한을 가진 코드워드를 생성하고 코드열 배치시 경계 규칙에 따라 코드워드가 대체되는 경우에도 최초의 코드열 특성을 그대로 유지하도록 코드워드를 배치하는 코드 배치 방법을 제공하는 데 있다.

【발명의 구성 및 작용】

<25> 본 발명에 따라, 상기 목적은 m비트의 소스데이터를 최소 구속장 d와 최대 구속장 k로 제한하면서 n비트($n \geq m$)의 코드워드로 변환하는 데이터 변조 방법에 있어서: (a) 입력 데이터열을 일정 길이로 분할하고, 상기 분할된 입력 데이터열을 다중화 정보를 이용

하여 소정의 다중화 방식에 의해 다중화해서 다중화된 데이터열을 제공하는 단계; 및
 (b) 상기 다중화된 데이터열에 대해 별도의 부가 비트가 부가된 DC 제어 변환표를 사용하지 않는 weak DC-free RLL(Run Length Limited) 변조를 수행하고, 다중화되고 RLL 변조된 코드열 중에서 가장 DC 성분이 작은 코드열을 제공하는 단계를 포함하는 데이터 변조 방법에 의해 달성된다.

<26> 본 발명의 다른 분야에 따르면, 상기 목적은 m비트의 소스데이터를 최소 구속장 d와 최대 구속장 k로 제한하면서 n비트($n \geq m$)의 코드워드로 변환하는 변조장치에 있어서: 다중화 정보를 이용하여 일정 길이로 분할된 입력 데이터를 다중화하여 다중화된 데이터열을 제공하는 다중화 수단; 상기 다중화된 데이터열에 대해 별도의 부가 비트가 부가된 DC 제어 변환표를 사용하지 않는 weak DC-free RLL(Run Length Limited) 변조를 수행하는 엔코딩 수단; 및 상기 다중화되고 RLL 변조된 코드열중에서 가장 DC성분이 작은 코드열을 선택하는 선택 수단을 포함하여, DC 억압 능력을 향상시킨 것을 특징으로 하는 데이터 변조 장치에 의해 달성된다.

<27> 본 발명의 또 다른 분야에 따르면, 상기 목적은 m비트의 소스데이터를 최소 구속장 $d=1$, 최대 구속장 $k=7$ 로 제한하면서 n비트($n \geq m$)의 코드워드들을 배열하는 코드 배치 방법에 있어서: 코드워드 a와 코드워드 b가 연결되고 코드워드 a가 선행하는 코드워드이며 코드워드 b는 코드워드 b1과 코드워드 b2중 선택이 가능하고 코드워드 a와 코드워드 b1이 연결되는 코드열을 코드열 X1, 코드워드 a와 코드워드 b2가 연결되는 코드열을 코드열 X2라 할 때 코드워드 b1과 b2는 코드워드내의 비트 "1"의 수가 홀수 인지 또는 짝수인지에 따라 다음 코드워드의 천이를 예측하는 INV

파라미터가 반대값을 갖도록 배치하는 단계; 및 코드워드 a와 코드워드 b1 또는 b2가 연결될 때 경계 규칙에 의해 코드워드 a 또는 코드워드 b1, b2가 다른 코드워드로 변환된다고 하더라도 코드열 X1과 X2는 그대로 INV 파라미터 값이 반대로 유지되도록 배치하는 단계를 포함하는 코드 배치 방법을 제공하는 데 있다.

- <28> 이하, 첨부된 도면을 참조하여 본 발명의 바람직한 실시 예를 설명하기로 한다.
- <29> 도 2는 본 발명에 의한 데이터 변조 장치의 일 실시 예에 따른 블록도이다.
- <30> 도 2를 참조하면, 입력 데이터열은 수학적 식 1에 도시된 바와 같이 $x=(x_0, x_1, \dots, x_{k-1})$ 로 표시할 수 있고, vXu 분할기(10)에서 입력 데이터열을 수학적 식 2에 도시된 바와 같이 $vXu(=k)$ 로 나누는 데, 즉 v바이트 단위의 입력 데이터열에 대해 u바이트 길이로 나눈다.

<31> 【수학적 식 1】 $\underline{x}=(x_0, x_1, \dots, x_{k-1}, \dots, x_{l-1})$

<32>

$$\underline{B_x} = \begin{bmatrix} x_{0,0} & x_{0,1} & \dots & x_{0,u-1} \\ x_{1,0} & x_{1,1} & \dots & x_{1,u-1} \\ \vdots & \vdots & & \vdots \\ x_{i,0} & x_{i,1} & \dots & x_{i,u-1} \\ \vdots & \vdots & & \vdots \\ x_{v-1,0} & x_{v-1,1} & \dots & x_{v-1,u-1} \end{bmatrix} = \begin{bmatrix} \underline{y_0} \\ \underline{y_1} \\ \vdots \\ \underline{y_i} \\ \vdots \\ \underline{y_{v-1}} \end{bmatrix}$$

【수학적 식 2】

<33> 여기서, $x_{i,j} = x_{ixu+j}$ 이다.

- <34> 다중화기(20)는 분할기(10)에 의해 분할된 vXu의 각각의 데이터열에 a비트의 부가 정보를 붙여서 $L=2^a$ 개의 데이터열로 다중화한 후 부가된 다중화 정보 's'에 따라 데이터열을 랜덤 데이터로 변환한다. 랜덤 데이터로의 변환이 끝나면 수학적 식 3 및 수학적 식 4와 같이 하나의 데이터열 y_i 에 대해 2^a 개로 다중화된 서로 다른 내용의 데이터가 만들어진다.

<35> 【수학식 3】 $\underline{C}_y = (\underline{C}_0, \underline{C}_1, \dots, \underline{C}_i, \dots, \underline{C}_{v-1})$

<36>
$$C_i = \begin{bmatrix} s_0, & y_{i,0}^0, & y_{i,1}^0, \dots, & y_{i,j}^0, \dots, & y_{i,u-1}^0 \\ s_1, & y_{i,0}^1, & y_{i,1}^1, \dots, & y_{i,j}^1, \dots, & y_{i,u-1}^1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ s_{L-1}, & y_{i,0}^{L-1}, & y_{i,1}^{L-1}, \dots, & y_{i,j}^{L-1}, \dots, & y_{i,u-1}^{L-1} \end{bmatrix} = \begin{bmatrix} f(y_i/s_0) \\ f(y_i/s_1) \\ \dots \\ f(y_i/s_{L-1}) \end{bmatrix}$$

【수학식 4】

<37> 여기서, 함수 $f(y_i/s)$ 는 다중화 정보 s 를 이용하여 y_i 에 대해 랜덤 데이터로 만든 결과를 의미한다.

<38> 동기 및 다중화 ID 삽입기(30)는 부가된 다중화 정보에 따른 복수(여기서는 $L=2^a$) 채널로 구성될 수 있으며, 2^a 개로 다중화된 랜덤 데이터열에 대해 즉, 다중화 정보가 부가된 다중화된 데이터열에 동기 패턴을 삽입하고, 다중화 정보를 다중화 식별자(ID)로 변환한다.

<39> 엔코더(40)는 부가된 다중화 정보에 따른 복수(여기서는 $L=2^a$) 채널로 구성될 수 있으며, 불충분한 DC 억압 코드에 대한 RLL(Run Length Limited) 변조를 수행한다. 본 발명의 엔코더(30)는 부가 비트가 부가된 별도의 DC 억압용 코드 변환표를 가지지 않아 리턴던시가 없으면 DC 억압은 가능하나 억압 성능은 떨어지는 코드를 사용하는 데, 일 실시 예로서, 엔코더(40)의 최소 구속장(d)=1, 최대 구속장(k)=7로 하여 그 예로 RLL(1,7,8,12) 코드를 사용할 수 있다. 동기 및 다중화 ID 삽입기(30)에서 다중화 정보를 다중화 ID로의 변환은 최소 구속장(d)=2, 최대 구속장(k)=7로 하여 최소 구속장을 키움으로서 최소 마크(또는 피트)의 크기를 키워 신호의 간섭 잡음을 줄일 수 있다. 다른 실시 예로서 엔코더(40)의 최소 구속장(d)=2, 최대 구속장(k)=10이 될 수 있으며, 그 예로 RLL(2,10,8,15) 코드를 들 수 있다.

- <40> 비교 및 선택기(50)는 RLL 변조된 2^a 개의 변조 스트림에 대해 DC 성분이 가장 적은 변조 스트림 하나를 선택한다.
- <41> 다음은 본 발명에서 제안하고 있는 weak dc-free RLL 변환 코드에 대해 설명한다.
- <42> (d,k,m,n) 으로 표현되는 RLL 코드에서 코드의 성능을 표현하는 요인 중에서 크게 기록 밀도의 측면과 DC 성분을 억압하는 능력을 보고, 그 코드의 우수함을 평가한다. 여기서, m 은 데이터 비트수(일명 소스 데이터의 비트수, 정보 워드 비트수라고도 함), n 은 변조 후의 코드워드 비트수(일명 채널 비트수라고도 함), d 는 코드워드 내에서 1과 1 사이에 존재할 수 있는 연속되는 0의 최소수(최소 구속장(Minimum Run Length Limit)이라고 함), k 는 코드워드내에서 1과 1사이에 존재할 수 있는 연속되는 0의 최대수(최대 구속장(Maximum Run Length Limit)이라고 함)이다. 코드워드내 비트 간격은 T (기록 또는 재생시 클럭 주기에 해당)로서 나타낸다.
- <43> 변조 방법에서 기록 밀도를 향상시킬 수 있는 방법은 d 와 m 은 주어진 조건으로 둔 채 코드워드의 비트수 n 을 줄이는 것이다. 그러나, RLL 코드는 코드워드내에서 최소 구속장 d 와 최대 구속장 k 를 만족해야 한다. 이 (d,k) 조건을 만족하면서 데이터 비트수가 m 이라고 할 때 $RLL(d,k)$ 를 만족하는 코드워드의 수는 2^m 개 이상이면 된다. 그러나, 실제 이러한 코드를 사용하기 위해서는 코드워드와 코드워드가 연결되는 부분에서도 런길이 제한 조건, 즉 $RLL(d,k)$ 조건을 만족해야 하며, 광디스크 기록/재생장치와 같이 코드의 DC 성분이 시스템 성능에 영향을 주는 경우에는 사용하고자 하는 코드가 DC 억압 능력을 가져야 한다.

- <44> 본 발명에서는 소스코드에 대해 변환될 코드워드의 코드표는 크게 2가지 즉, 1) 주 변환표, 2) DC 제어용 보조 변환표로서 생성된다.
- <45> 각각의 변환표내의 코드워드 생성방식은 아래와 같으며, 최소 구속장이 1이고 최대 구속장이 7인 (1,7) 코드를 예로 들어 설명하기로 한다.
- <46> 도 3은 주코드 변환표의 여러 코드워드 그룹과 해당 코드그룹의 코드워드 특성을 표로 나타낸 것이다.
- <47> 코드의 최소 구속장을 d , 최대 구속장을 k , 소스데이터의 비트수를 m , 변조후의 코드워드의 비트수를 n 이라 하고 코드워드의 LSB(Least Significant Bit)로부터 MSB(Most Significant Bit)방향으로 연속하는 0의 수를 EZ(End Zero), MSB로부터 LSB방향으로 연속하는 0의 수를 LZ(Lead Zero)라 할 때 $d=1$, $k=7$, $m=8$, $n=12$, $0 \leq EZ \leq 5$ 인 코드워드를 LZ의 조건에 따라 분류하면 다음과 같다.
- <48> (1) $1 \leq LZ \leq 7$ 를 만족하는 코드워드의 수 : 210개
- <49> (2) $0 \leq LZ \leq 4$ 를 만족하는 코드워드의 수 : 316개
- <50> (3) $0 \leq LZ \leq 2$ 를 만족하는 코드워드의 수 : 264개
- <51> $m=8$ 인 소스데이터의 변조를 위해서는 코드워드의 수가 최소 256개 이상이어야 하지만, 위에서 (1)의 경우 코드워드의 수가 256개에 미치지 못하므로 다른 LZ 조건을 만족하는 코드워드 중에서 일부를 가져와 필요한 코드워드의 수를 만족시키도록 할 수 있다. 이 경우 상기 (2)에서 $LZ=0$ 인 코드워드중에서 1010xxxxxxx인 51개를 빼서 상기 (1)의 그룹에 추가하면 상기 (1) 그룹에 속하는 코드워드의 수는 261개가 되며 상기 (2)의 그룹에 속하는 코드워드의 수는 $316-51=265$ 개, 상기 (3)의 그룹에 속하는 코드워드의 수는

264개로 (1)~(3) 그룹 모두 256개 이상의 코드워드를 보유하게 되어 각 조건에 해당하는 그룹의 코드워드의 수가 8비트의 소스 데이터에 대해 256개라는 변조 코드워드의 최소 개수를 만족시킬 수 있다. 이 중 각각 256개의 코드워드만을 골라 3개의 주코드 변환표(MCG1~MCG3)를 만든다. 도 3의 표에서 MCG(Main Code Group)1은 상기 (1)의 조건에 해당하는 코드워드들과 (2)를 만족하는 그룹으로부터 가져온 일부(51개)의 코드워드들을 포함하는 그룹의 명칭이고, MCG2와 MCG3는 각각 차례로 상기 (2), (3)의 조건에 해당하는 코드워드들을 포함하는 그룹의 명칭이며, 이들 주코드 그룹들(MCG1-MCG3) 각각으로부터 256개의 코드워드들만이 소스코드에 대한 변환 코드들로서 사용될 수 있다.

<52> 도 4는 DC 제어용 보조 변환표의 여러 코드워드 그룹과 해당 코드그룹의 코드워드 특성을 표로 나타낸 것이다.

<53> DC 제어용 보조 변환표내의 코드워드는 $d=1$, $k=7$, $m=8$, $n=12$ 인 코드워드중 $6 \leq EZ \leq 7$ 를 만족하는 코드워드와 주코드 그룹(Main Code Group)에서 남는 코드워드들과 $LZ=5$, 6 또는 $LZ=3$ 인 코드워드를 합쳐 DC 제어용 보조 코드그룹(ACG)의 코드워드로서 사용한다. 이 코드워드의 생성 조건을 구체적으로 설명하면 다음과 같으며, 각 항목은 차례로 도 4의 표에서 DC 제어용 보조 변환표의 이름은 ACG1, ACG2, ACG3로서 나타내어지고 있다.

<54> ACG1: $6 \leq EZ \leq 7$ 이고 $LZ \neq 0$ 을 만족하는 코드워드 8개 + MCG1에서 남는 코드워드 5개 + $6 \leq EZ \leq 7$ 이고 $LZ=0$ 이면서 1010xxxxxxx인 코드워드 2개 = 15개

<55> ACG2: $6 \leq EZ \leq 7$ 이고 $0 \leq LZ \leq 6$ 을 만족하는 코드워드 12개 + $0 \leq EZ \leq 5$ 이고 $5 \leq LZ \leq 6$ 을 만족하는 코드워드 21개 + MCG2에서 남는 코드워드 9개 - $6 \leq EZ \leq 7$ 이고 $LZ=0$ 이면서 1010xxxxxxx인 코드워드 2개 = 40개

- <56> ACG3: $6 \leq EZ \leq 7$ 이고 $0 \leq LZ \leq 3$ 을 만족하는 코드워드 10개 + $0 \leq EZ \leq 5$ 이고 $LZ=3$ 을 만족하는 코드워드 33개 + MCG3에서 남는 코드워드 8개 = 51개
- <57> 도 5는 도 3에서 설명한 주 변환표와 도 4에 도시된 DC 제어용 보조 변환표에서 이전 코드워드 a의 엔드 제로(EZ_a)에 따라 결정되는 다음 코드 그룹($M=ncgdet$) 즉, 이전 코드워드 a의 다음 코드 그룹을 나타내는 파라미터 ncg를 정리한 표이다. 코드워드 b는 이전 코드워드 a의 엔드 제로수(EZ_a)에 따라 그 코드워드 b가 속해있는 코드그룹이 정해지게 되는 데, EZ_a가 0이면 다음 코드워드 b가 속해있는 코드그룹은 1(=MCG1)이고, EZ_a가 1이상이고 3이하이면 다음 코드워드 b가 속해있는 코드그룹은 2(=MCG2)이고, EZ_a가 4이상이고 7이하이면 코드워드 b가 속해있는 코드그룹은 3(=MCG3)이다.
- <58> 한편, 코드워드 a와 코드워드 b가 연결되는 지점에서도 구속장 (d, k)조건을 만족해야한다. 도 6은 코드워드 a와 b가 연결될 때 구속장의 조건을 위해 고려해야 할 것을 보인 것이다. 도 6에서 코드워드 a의 엔드 제로(EZ_a)와 코드워드 b의 리드 제로(LZ_b)를 더한 값이 최소 런 길이 d 이상이고 최대 런 길이 k 이하여야 한다는 것이 구속장의 조건을 만족하는 것이라 할 수 있다.
- <59> 도 7은 도 6을 통해 설명한 구속장의 조건을 만족하지 못하는 경우가 발생하는 경우, 코드 변환 전과 후에 따른 파라미터 INV의 변화를 보이고 있다. 파라미터 INV는 다음 코드워드의 천이를 알 수 있는 파라미터로서, 코드워드내에서 비트 "1"의 수가 짝수개이면 INV값은 "0"이고, 코드워드내에서 비트 "1"의 수가 홀수개이면 INV의 값은 "1"이다. 부가적으로, DSV(Digital Sum Value)는 코드워드 스트림에서 디지털 합 값으로서, DSV의 절대치가 작으면 직류 성분 또는 저주파 성분이 적음을 의미한다. CSV는 코드워드내에서 디지털 합 값으로서 CSV가 작으면 코드워드의 직류 성분 또는 저주파 성분이 적

음이 의미하며, 코드 변환중에 직류 성분이나 저주파 성분의 평가에는 CSV가 이용된다. 코드워드 스트림에서 현 코드워드까지 누적된 INV의 값이 "0"이면 다음 코드워드의 CSV값을 그 코드워드 이전까지의 누적된 DSV값에 그대로 더하여 DSV값을 갱신하고, 누적된 INV값이 "1"이면 다음 코드워드의 CSV값의 부호를 반전시켜 그 코드워드 이전까지의 누적된 DSV값에 더하여 DSV값을 갱신한다.

<60> 도 7을 참조하면, 코드워드 b는 이전 코드워드 a의 EZ에 따라 그 코드워드 b가 속해 있는 코드그룹이 정해지게 되는데, 주 변환표와 DC 제어용 보조 변환표에서 코드워드 수가 부족하여 다른 코드 변환표에서 코드워드를 빌려온 코드그룹이 지정되는 경우에 (d,k) 조건을 만족하지 못하는 경우가 발생한다. 도 7에서는 $d \leq EZ_a + LZ_b \leq k$ 를 위반하는 경우를 예시하였으며 이 경우에는 코드워드 a의 EZ가 바뀌게 되는데 이렇게 구속장 조건을 만족하지 않아 코드워드 변화가 일어나게 되는 것을 경계 규칙이라 한다. 코드워드 스트림내의 비트 "1"의 수가 짝수인지 홀수인지를 나타내는 파라미터 INV는 경계 규칙에 의해 코드 변화 전의 상태로부터 변할 가능성이 있다. 이런 특징때문에 코드워드를 배치할 때 특히 DC 제어가 가능한 코드 변환표끼리는 주의가 필요하다.

<61> 도 8은 DC 제어 가능한 코드워드 b1, b2로 인한 코드열의 분기 예를 도시한 것이다. 본 발명의 코드 변환에 있어 가장 큰 특징중 하나는 DC 제어를 하기 위해 선택 가능한 두개의 코드 변환표내의 코드워드들은 INV(코드워드 스트림내의 비트 "1"의 수가 짝수 또는 홀수를 나타냄) 특징을 반대로 유지하도록 되어 있다. 이렇게 하면 두 개의 코드변환표내의 코드워드들은 INV가 반대이므로 두 코드워드 중 하나는 DC 제어가 최적인 방향으로 진행될 수 있기 때문이다. 그러나 앞서 경계 규칙에서 설명했듯이 INV에 변화가 발생하는 경우가 있는데 이 경우는 DC 제어가 가능한 시점에서의 선택가능한 두개

의 코드 변환표에 모두 같은 현상 즉, INV가 두개의 선택가능한 코드 변환표에서 똑같이 변화되면 아무런 문제가 없다. 이를 위해 본 발명에서는 다음과 같은 사항을 고려해서 코드 변환표를 설계하였다.

<62> 먼저, 도 8의 A, 즉, 코드워드 a와 코드워드 b가 연결되는 지점에서 코드워드 b로서 선택가능한 b1과 b2인 경우, EZ_a가 "xxxxxxxxx101"일 때, LZ_b1(코드워드 b1의 리드 제로수)가 "101xxxxxxxxx"이면, 그때 LZ_b2(코드워드 b2의 리드 제로 수)가 각각 "101xxxxxxxxx"가 된다. 즉, 리드 제로수가 "101xxxxxxxxx"인 코드워드들은 MCG1과 ACG1내의 같은 위치에 배치하고, EZ가 "xxxxxxxxx101"인 모든 코드들은 각각 MCG1과 ACG1, MCG2와 ACG2, MCG3와 ACG3내의 같은 위치에 배치하여 코드워드 a의 엔드 제로수가 "xxxxxxxxx101"인 경우 경계 규칙에 의해 코드워드 b1이 속해있는 코드열이나 코드워드 b2가 속해있는 코드열 모두 코드워드 a의 INV가 바뀌거나 또는 모두 바뀌지 않아 결국 두 코드열의 INV는 반대로 유지하도록 한다.

<63> 다음으로, 도 8의 B 즉, 코드워드 b와 코드워드 c가 연결되는 점에서 코드워드 b1과 b2가 각각 코드워드 c와 연결될 때 경계 규칙에 의해 코드워드 b1, b2 또는 코드워드 c가 다른 코드워드로 변환된다고 하더라도 코드워드 b1과 코드워드 c가 연결되는 코드열과 코드워드 b2와 코드워드 c가 연결되는 코드열은 그대로 파라미터 INV값이 반대로 유지된다.

<64> 동기 패턴 및 다중화 ID에 대해 설명한다.

<65> 비트 1과 1사이의 0의 최대수 k=7로 제한된 변조 방식에서 동기 패턴으로 k=7의 제한을 위반하는 "010000000010000000010"을 사용하기로 한다.

<66> Sync Pattern : 010000000010000000010

<67> 다중화 ID는 데이터열의 다중화를 위해 4비트의 다중화 정보를 이용해서 도 9에 도시된 바와 같이 6비트로 변환되고, 이 경우 하나의 데이터열은 $L=2^4=16$ 종류의 랜덤 데이터열로 변환된다.

<68> 도 10a 내지 도 10e는 상술한 사항들을 고려하여 생성 및 배치한 주 변환 코드표이다.

<69> 도 11은 상술한 사항들을 고려하여 생성 및 배치한 보조 DC 제어용 코드 변환표이다. DC 제어용 보조 변환표내의 코드워드를 사용할 때에는 이전 코드워드(도 8의 코드워드 a)와 뒤에 따라오는 코드워드(도 8의 코드워드 c)와의 런 길이 위반 여부를 검사한 뒤 위반하지 않는 경우에만 사용하여야 한다.

<70> 도 12는 본 발명에서 제안한 방식대로 RLL(1,7) 변조를 수행한 후 코드열에 대한 PSD 곡선을 보인 도면이다.

<71> 도 13은 본 발명에서 제안한 RLL(1,7) 코드와 종래의 코드들(A-Code, B-Code, C-Code)과 비교하여 얻어진 코드율과 코드효율을 정리한 표이다. 본 발명의 RLL(1,7) 코드는 종래의 코드열과 비교하여 유사한 DC 억압능력을 가지면서 코드의 효율이 높고 기록밀도를 약 2% 올릴 수 있는 장점이 있다.

<72> 도 14는 RLL(2,10,8,15) 코드를 본 발명에 적용하였을 경우 종래의 DVD에서 사용되고 있는 EFMP(Eight to Fourteen Modulation Plus) 코드와 DC 억압 능력을 비교한 PSD

곡선을 보이고 있다. 여기서, EFMP 코드는 주 변환 코드표와는 별도의 DC 제어용 코드 변환표를 사용해서 DC 억압을 행하는 코드이다. RLL(2,10,8,15) 코드의 일 예는 동일 출원인에 의해 출원된 국내 출원 번호 P2001-0021360호에 개시되어 있으며, 본 발명에서는 부가 비트를 사용한 DC 억압용 코드 변환표를 사용하지 않고 주 변환표와 DC 제어용 보조 변환표를 사용하는 불충분한 DC 억압 RLL 변조를 수행하고 있다.

<73> 도 15는 종래의 EFMP 코드와 본 발명이 적용된 RLL(2,10,8,15) 코드를 비교하여 코드율과 코드 효율을 정리한 표이다. 본 발명에서 제안한 방식의 RLL(2,10,8,15) 코드가 EFMP와 비교해서도 유사한 DC 억압 능력을 가지면서 높은 코드효율을 갖고 기록밀도도 약 5.4% 향상된다.

【발명의 효과】

<74> 상술한 바와 같이, 본 발명은 불충분한 DC 억압 변조 코드에 대해 멀티모드 코딩 방식을 결합하여 DC 억압 능력을 향상시킨 고효율의 변조 코드를 제공함으로써 기록밀도 측면에서 높은 효율성을 제공한다.

<75> 또한, 본 발명은 불충분한 DC 억압 RLL 변조시 코드워드들 사이에 구속장 조건을 만족하지 않아 코드워드를 다른 코드워드로 대체하는 경우에도 코드열의 DC 억압 능력을 유지하도록 코드워드를 배치함으로써 코드열의 보다 나은 DC 억압 능력을 제공한다.

【특허청구범위】**【청구항 1】**

m비트의 소스데이터를 최소 구속장 d와 최대 구속장 k로 제한하면서 n비트($n \geq m$)의 코드워드로 변환하는 데이터 변조 방법에 있어서:

(a) 입력 데이터열을 일정 길이로 분할하고, 상기 분할된 입력 데이터열을 다중화 정보를 이용하여 소정의 다중화 방식에 의해 다중화해서 다중화된 데이터열을 제공하는 단계; 및

(b) 상기 다중화된 데이터열에 대해 별도의 부가 비트가 부가된 DC 제어 변환표를 사용하지 않는 weak DC-free RLL(Run Length Limited) 변조를 수행하고, 다중화되고 RLL 변조된 코드열 중에서 가장 DC 성분이 작은 코드열을 제공하는 단계를 포함하는 데이터 변조 방법.

【청구항 2】

제1항에 있어서, 상기 다중화 방식은 스크램블 방식을 이용하는 것을 특징으로 하는 데이터 변조 방법.

【청구항 3】

제1항에 있어서, 상기 다중화 방식은 인터리브 방식을 이용하는 것을 특징으로 하는 데이터 변조 방법.

【청구항 4】

제1항에 있어서, 상기 (b) 단계에서는 소정 구속장 조건에 맞는 코드워드를 생성하고, 상기 구속장 조건별로 코드워드들로 그룹핑하고, 소스워드에 대한 코드열이 DC 제어

능력을 갖도록 코드워드들을 배치되어 있는 주코드 변환표와 상기 소정 구속장 조건을 만족하고, 상기 주 변환 코드워드 테이블에서 필요하지 않은 코드워드들을 가져와 DC 제어용 보조 변환표를 이용하여 RLL 변조를 수행하는 것을 특징으로 하는 데이터 변조 방법.

【청구항 5】

제4항에 있어서, 최소 구속장(d)=1, 최대 구속장(k)=7인 것을 특징으로 하는 데이터 변조 방법.

【청구항 6】

제4항에 있어서, 상기 (b) 단계에서 DC 제어를 위해 선택 가능한 두 종류의 코드그룹 즉, 주 변환 코드그룹과 DC 제어용 보조 변환 코드그룹내의 동일 데이터에 대한 코드워드가 서로 반대의 INV값을 갖는 특징을 가지고, INV는 코드워드 스트림내에서 비트 "1"의 수가 홀수인지 짝수인지에 따라 다음 코드워드의 천이 방향을 예측하는 파라미터인 것을 특징으로 하는 데이터 변조 방법.

【청구항 7】

제4항에 있어서, 상기 주코드 변환표는 코드워드의 LSB(Least Significant Bit)로부터 MSB(Most Significant Bit)방향으로 연속하는 0의 수를 나타내는 엔드 제로(EZ)가 0이상 5이하에 있으면서 각각 코드워드의 MSB로부터 LSB 방향으로 연속하는 0의 수를 나타내는 리드 제로(LZ)가 1이상 7이하, 0이상 4이하, 0이상 2이하의 조건을 만족하는 코드워드들의 그룹으로 이뤄짐을 특징으로 하는 데이터 변조 방법.

【청구항 8】

제7항에 있어서, 상기 주코드 변환표의 그룹 중 소스워드가 변환가능한 최소한의 코드워드 개수보다 적은 코드워드로 이뤄진 그룹은 상기 최소한의 코드워드 개수보다 많은 코드워드를 가진 그룹으로부터 잉여 코드를 가져와 최소 코드워드 개수 이상의 코드워드를 가지도록 하는 것을 특징으로 하는 데이터 변조 방법.

【청구항 9】

제4항에 있어서, 상기 DC 제어용 보조 변환표는

EZ가 6이상이고 7이하이면서 LZ가 0이 아닌 코드워드, 제1 주 변환 코드 그룹의 잉여 코드워드와 EZ가 6이상이고 7이하이면서 LZ가 0이면서 "1010xxxxxxx"인 코드워드를 가져와 만든 그룹,

EZ가 6이상이고 7이하이면서 LZ가 0이상이고 6이하인 코드워드, EZ가 0이상이고 5이하이면서 LZ가 5이상이고 6이하를 만족하는 코드워드, 제2 주 변환 코드 그룹의 잉여 코드와의 합에서 EZ가 6이상이고 7이하이면서 LZ가 0이면서 "1010xxxxxxx"인 코드워드를 빼서 만든 그룹, 및

EZ가 6이상이고 7이하이면서 LZ가 0이상이고 3이하를 만족하는 코드워드, EZ가 0이상이고 5이하이고 LZ가 3을 만족하는 코드워드와 제3 주 변환 코드 그룹의 잉여 코드를 가져와 만든 그룹으로 구성함을 특징으로 하는 데이터 변조 방법.

【청구항 10】

제9항에 있어서,

코드열쌍이 차례로 a, b_1, c 인 것과 a, b_2, c 이고, b_1 과 b_2 는 서로 다른 특성의 INV를 가지는 상기 DC 제어용 보조 변환표내의 코드워드일 때, 상기 a 와 b_1, b_2 그리고 b_1, b_2 와 c 사이에 소정 런 길이 위반(경계 조건 위반)이 발생되어 a, b_1, b_2 또는 c 의 코드 변환이 일어나더라도 변환 후의 코드열 a, b_1, c 와 a, b_2, c 의 INV 특성이 서로 반대를 유지하도록 코드워드가 상기 주 변환표와 DC 제어용 보조 변환표에 배치되어 있는 것을 특징으로 하는 데이터 변조 방법.

【청구항 11】

제4항에 있어서, 최소 구속장(d)=2, 최대 구속장(k)=10인 것을 특징으로 하는 데이터 변조 방법.

【청구항 12】

제1항에 있어서, 상기 방법은

(c) 상기 다중화 정보가 부가된 다중화된 데이터열에 동기 패턴을 삽입하고, 상기 다중화 정보를 다중화 식별자정보(ID)로 변환하는 단계를 더 포함하는 데이터 변조 방법

【청구항 13】

제12항에 있어서, 상기 다중화 ID는 최소 구속장(d)=2, 최대 구속장(k)=7로서, 최소 구속장을 키워서 최소 마크(또는 피트)의 크기를 키움으로서 신호의 간섭 잡음을 줄이는 것을 특징으로 하는 데이터 변조 방법.

【청구항 14】

m비트의 소스데이터를 최소 구속장 d와 최대 구속장 k로 제한하면서 n비트($n \geq m$)의 코드워드로 변환하는 데이터 변조장치에 있어서:

다중화 정보를 이용하여 일정 길이로 분할된 입력 데이터를 다중화하여 다중화된 데이터열을 제공하는 다중화 수단;

상기 다중화된 데이터열에 대해 별도의 부가 비트가 부가된 DC 제어 변환표를 사용하지 않는 weak DC-free RLL(Run Length Limited) 변조를 수행하는 엔코딩수단; 및

상기 다중화되고 RLL 변조된 코드열중에서 가장 DC성분이 작은 코드열을 선택하는 선택 수단을 포함하여,

DC 억압 능력을 향상시킨 것을 특징으로 하는 데이터 변조 장치.

【청구항 15】

제14항에 있어서,

입력 데이터열을 일정 길이로 분할하는 분할 수단; 및

상기 다중화 정보가 부가된 다중화된 데이터열에 동기 패턴을 삽입하고 상기 다중화 정보를 다중화 식별자정보(ID)로 변환하는 동기 및 다중화 ID 삽입 수단을 더 포함하는 데이터 변조 장치.

【청구항 16】

제14항에 있어서, 상기 다중화 수단은 분할된 입력 데이터를 스크램블 방식을 이용하여 랜덤 데이터로 변환하는 것을 특징으로 하는 데이터 변조 장치.

【청구항 17】

제14항에 있어서, 상기 다중화 수단은 분할된 입력 데이터를 인터리브 방식을 이용하여 랜덤 데이터로 변환하는 것을 특징으로 하는 데이터 변조 장치.

【청구항 18】

제14항에 있어서, 상기 엔코딩 수단은 DC 제어를 위해 선택 가능한 두 종류의 코드 그룹 즉, 주 변환 코드그룹과 DC 제어용 보조 변환 코드그룹내의 동일 데이터에 대한 코드워드가 서로 반대의 INV값을 갖는 특징을 가지고, INV는 코드워드 스트림내에서 비트 "1"의 수가 홀수인지 짝수인지에 따라 다음 코드워드의 천이 방향을 예측하는 파라미터인 것을 특징으로 하는 데이터 변조 장치.

【청구항 19】

제14항에 있어서, 상기 엔코딩 수단은 최소 구속장(d)=1, 최대 구속장(k)=7인 것을 특징으로 하는 데이터 변조 장치.

【청구항 20】

제19항에 있어서, 상기 소스워드의 비트 길이가 8일 때, 상기 변조 코드워드의 길이는 12비트인 것을 특징으로 하는 데이터 변조 장치.

【청구항 21】

제15항에 있어서, 상기 동기 및 다중화 ID 삽입 수단은 최소 구속장(d)=1, 최대 구속장(k)=7인 것을 특징으로 하는 데이터 변조 장치.

【청구항 22】

제15항에 있어서, 상기 엔코딩 수단은 최소 구속장(d)=1, 최대 구속장(k)=7이고, 상기 동기 및 다중화 ID 삽입 수단은 최소 구속장(d)=2, 최대 구속장(k)=7로서 최소 구속장을 키움으로서 최소 마크(또는 피트)의 크기를 키워 신호의 간섭 잡음을 줄이도록 한 것을 특징으로 하는 데이터 변조 장치.

【청구항 23】

제15항에 있어서, 상기 엔코딩 수단은 최소 구속장(d)=2, 최대 구속장(k)=10인 것을 특징으로 하는 데이터 변조 장치.

【청구항 24】

제23항에 있어서, 상기 소스워드의 비트 길이가 8일 때, 상기 변조 코드워드의 길이는 15비트인 것을 특징으로 하는 데이터 변조 장치.

【청구항 25】

제23항에 있어서, 상기 동기 및 다중화 ID 삽입 수단은 최소 구속장(d)=2, 최대 구속장(k)=10인 것을 특징으로 하는 데이터 변조 장치.

【청구항 26】

m 비트의 소스데이터를 최소 구속장 $d=1$, 최대 구속장 $k=7$ 로 제한하면서 n 비트($n \geq m$)의 코드워드들을 배열하는 코드 배치 방법에 있어서:

코드워드 a 와 코드워드 b 가 연결되고 코드워드 a 가 선행하는 코드워드이며 코드워드 b 는 코드워드 b_1 과 코드워드 b_2 중 선택이 가능하고 코드워드 a 와 코드워드 b_1 이 연결되는 코드열을 코드열 X_1 , 코드워드 a 와 코드워드 b_2 가 연결되는 코드열을 코드열 X_2 라

할 때 코드워드 b1과 b2는 코드워드내의 비트 "1"의 수가 홀수 인지 또는 짝수인지에 따라 다음 코드워드의 천이를 예측하는 INV 파라미터가 반대값을 갖도록 배치하는 단계; 및

코드워드 a와 코드워드 b1 또는 b2가 연결될 때 경계 규칙에 의해 코드워드 a 또는 코드워드 b1, b2가 다른 코드워드로 변환된다고 하더라도 코드열 X1과 X2는 그대로 INV 파라미터 값이 반대로 유지되도록 배치하는 단계를 포함하는 코드 배치 방법.

【청구항 27】

제26항에 있어서, 코드워드 a와 코드워드 b1, b2사이의 비트 0의 수가 $d(=1)$ 보다 작을 경우에 경계 규칙을 두어 코드워드 a또는 코드워드 b1, b2를 변형함으로서 변형된 코드워드사이의 비트 "0"의 수가 $d(=1)$ 보다 크거나 같고 $k(=7)$ 보다 작거나 같도록 배치된 것을 특징으로 하는 코드 배치 방법.

【청구항 28】

제26항에 있어서, 상기 경계 규칙에 의해 상기 코드열 X1내의 코드워드 a와 코드열 X2내의 코드워드 a가, a와는 다른 코드워드로 변환되고 그러나 각각의 코드열에서 변환된 코드워드는 동일한 INV값을 갖도록 하여 코드워드 b1과 b2의 INV에 의해 결국 각각의 코드열 X1과 X2는 다른 INV값을 갖도록 배치된 것을 특징으로 하는 코드 배치 방법.

【청구항 29】

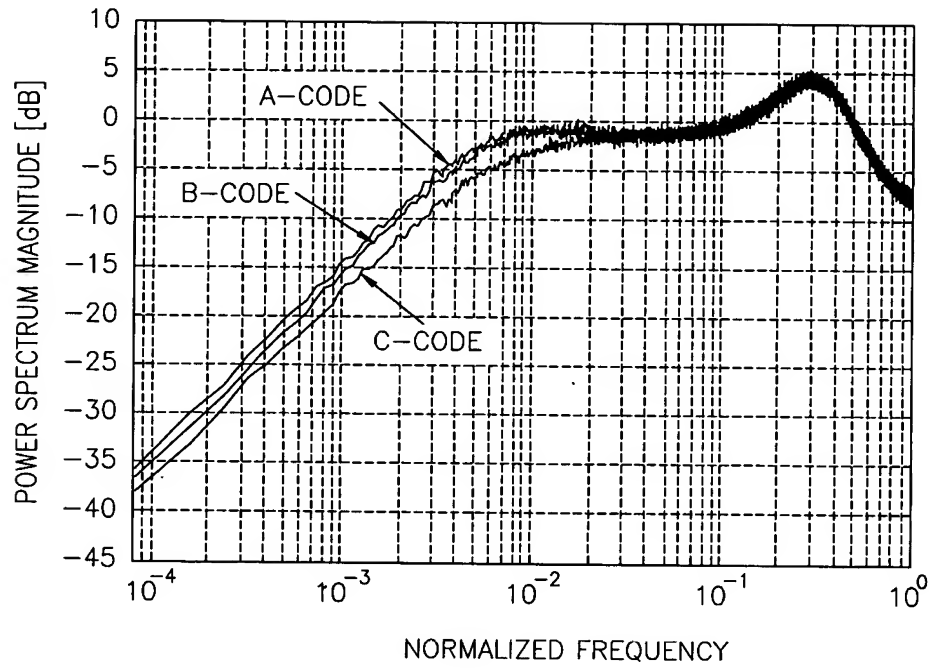
m 비트의 소스데이터를 최소 구속장 $d=1$, 최대 구속장 $k=7$ 로 제한하면서 n 비트($n \geq m$)의 코드워드들을 배열하는 코드 배치 방법에 있어서

코드워드 b와 코드워드 c가 연결되고 코드워드 b가 선행하는 코드워드이며 코드워드 b는 코드워드 b1과 코드워드 b2중 선택이 가능하고 코드워드 b1과 코드워드 c가 연결되는 코드열을 코드열 Y1, 코드워드 b2와 코드워드 c가 연결되는 코드열을 코드열 Y2라 할 때 코드워드 b1과 b2는 코드워드내의 비트 "1"의 수가 홀수 인지 또는 짝수인지에 따라 다음 코드워드의 천이를 예측하는 INV 파라미터가 반대값을 갖도록 배치된 단계; 및

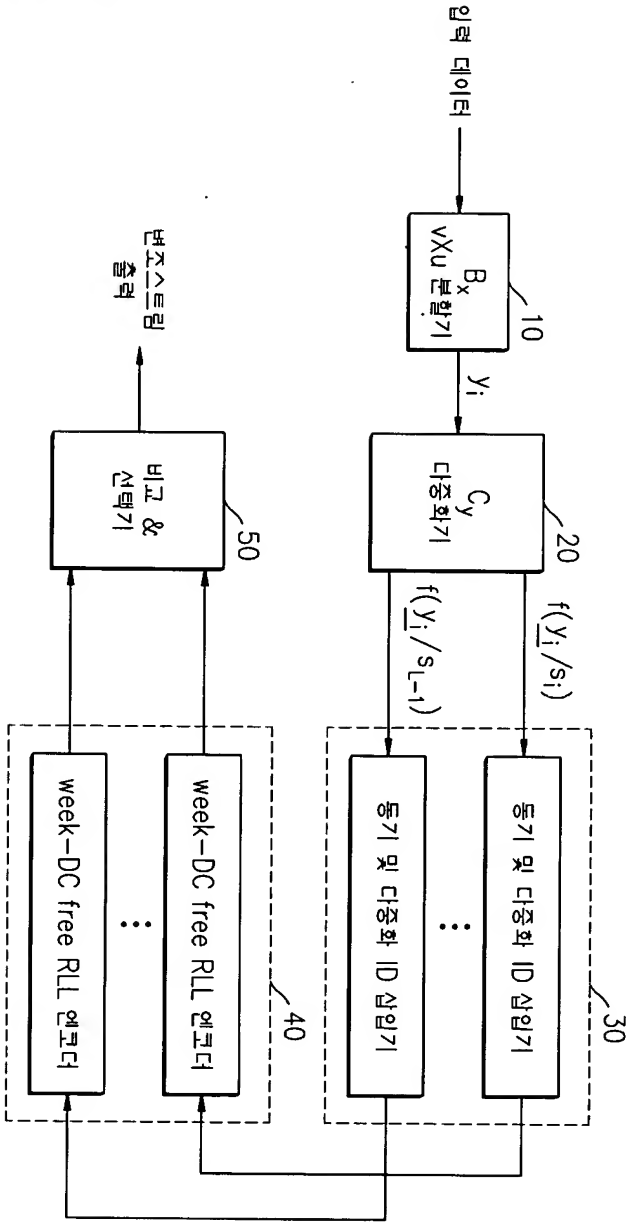
코드워드 b1 또는 b2가 코드워드 c와 연결될 때 경계규칙에 의해 코드워드 b1, b2 또는 코드워드 c가 다른 코드워드로 변환된다고 하더라도 코드열 Y1과 Y2는 그대로 INV 파라미터 값이 반대로 유지되도록 배치된 단계를 포함하는 코드 배치 방법.

【도면】

【도 1】



【도 2】



【표 3】

	LZ (EZ)	Number	Supplement	Number	Elimination	Number	Total Number (Redundant)
MCg1	1≤LZ≤7 (0≤EZ≤5)	210	1010xxxxxxx (0≤EZ≤5)	51			261(5)
MCg2	0≤LZ≤4 (0≤EZ≤5)	316			1010xxxxxxx (0≤EZ≤5)	51	265(9)
MCg3	0≤LZ≤2 (0≤EZ≤5)	264					264(8)

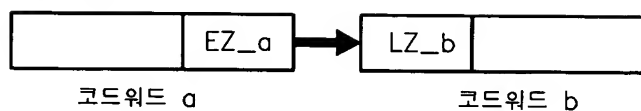
【표 4】

	LZ (EZ)	Number	Supplement 1	Number	Supplement 2	Number	Supplement 3 /Elimination	Number	Total Number
ACG1	$1 \leq LZ \leq 7$ ($6 \leq EZ \leq 7$)	8			Redundant Code Word from MCG1	5	1010xxxxxxx ($6 \leq EZ \leq 7$)	2	15
ACG2	$0 \leq LZ \leq 6$ ($6 \leq EZ \leq 7$)	12	$5 \leq LZ \leq 6$ ($0 \leq EZ \leq 5$)	21	Redundant Code Word from MCG2	9	1010xxxxxxx ($6 \leq EZ \leq 7$)	-2	40
ACG3	$0 \leq LZ \leq 3$ ($6 \leq EZ \leq 7$)	10	$LZ=3$ ($0 \leq EZ \leq 5$)	33	Redundant Code Word from MCG3	8			51

【표 5】

EZ_a	M=ncgdet (코드워드 a)
0	1
1-3	2
4-7	3

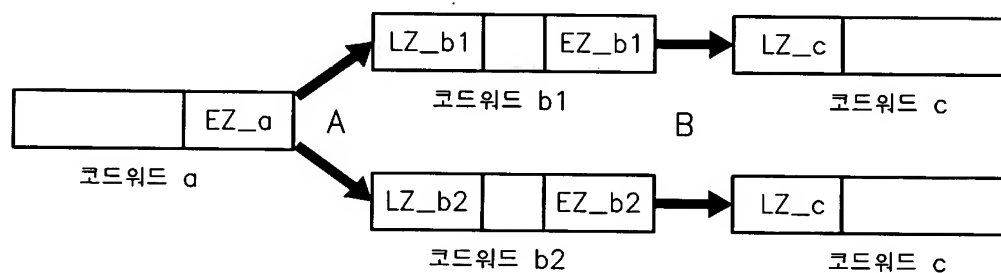
【도 6】



【도 7】

코드워드 a	코드워드 b	INV 의 변화
xxxxxxxx0101 → xxxxxxxx0100	1010xxxxxxxx	있음
xxxxxxxx01001 → xxxxxxxx01010		없음
~		
xxx100000001 → xxx100001010		

【도 8】



【도 9】

다중화 정보	변환된 다중화 ID	NEXT STATE
0	100001	1
1	100010	2
2	100100	2
3	100101	1
4	101000	2
5	101001	1
6	000100	2
7	000101	1
8	001000	2
9	001001	1
A	001010	2
B	010000	3
C	010001	1
D	010010	2
E	010100	2
F	010101	1

【도 10a】

Data Symbol	MCG1			MCG2			MCG3		
	Code Word MSB	LSB	NCG	Code Word MSB	LSB	NCG	Code Word MSB	LSB	NCG
000	000001010000		3	000010010000		3	001000001001		1
001	000010010000		3	000010001001		1	001000001010		2
002	000000100000		3	000010001010		2	001000010010		2
003	000101010000		3	000010010010		2	001000010101		1
004	001001010000		3	000010010101		1	001000010100		2
005	001010010000		3	000010010100		2	001000101001		1
006	001010100000		3	000010000001		1	001000000010		2
007	000010100000		3	000010000010		2	001001010000		3
008	101001010000		3	000010100000		3	001001001001		1
009	101010010000		3	000010010001		1	001000000100		2
010	000100010000		3	000010100010		2	001000001000		2
011	000000101001		1	000010100100		2	001000000101		1
012	000000101010		2	000010100101		1	001010010000		3
013	000001001010		2	000010101000		2	001001010001		1
014	000001010101		1	000010101001		1	001000101010		2
015	010001010000		3	000010000100		2	001001001010		2
016	000000100001		1	000101010000		3	001001010101		1
017	000000100010		2	000100000001		1	001001010010		2
018	000000100100		2	000010001000		2	001000010001		1
019	000000100101		1	000010101010		2	001000100010		2
020	000001001001		1	000010000101		1	001010100000		3
021	000001010010		2	000100010000		3	001010001001		1
022	000001010100		2	001001010000		3	001001010100		2
023	000010010101		1	001010010000		3	001010001010		2
024	000010001010		2	001010100000		3	001010010101		1
025	000001000001		1	010001010000		3	001010010010		2
026	000000101000		2	010010010000		3	001000100001		1
027	010010010000		3	000100100000		3	001000100100		2
028	000010000001		1	010010100000		3	001000010000		3
029	000001000010		2	010100010000		3	001001000001		1
030	000001000100		2	001000010000		3	001000101000		2
031	000001000101		1	000100010101		1	001001000010		2
032	010010100000		3	000100001010		2	001000100101		1
033	000010101001		1	000100101001		1	010001010000		3
034	000001001000		2	000100000010		2	010010010000		3
035	000010000010		2	010100100000		3	010010100000		3
036	000010000101		1	000101001001		1	010100010000		3
037	000010000100		2	000100000100		2	001000100000		3
038	000001010001		1	000100001000		2	010100100000		3
039	000010010010		2	000100000101		1	100001010000		3
040	000010001001		1	000101010001		1	010000010000		3
041	000010010100		2	000100101010		2	010000100000		3
042	000010100010		2	000101001010		2	010101010000		3
043	000010100101		1	000101010101		1	100010010000		3
044	000010100100		2	000101010010		2	001010010001		1
045	000100000001		1	000010100001		1	001010010100		2
046	000010001000		2	000100010010		2	001010100010		2
047	010100010000		3	001000100000		3	001010100101		1
048	000100101001		1	000100001001		1	001010100100		2
049	000010101010		2	000100010100		2	001010000001		1
050	000100000010		2	000100100010		2	001001000100		2
051	000100000101		1	000100100101		1	001001001000		2
052	010100100000		3	000100100100		2	001001000101		1
053	000101001001		1	000100101000		2	001010000010		2
054	000100000100		2	000101000101		1	001010100001		1
055	000100001000		2	000101000010		2	001010101000		2
056	000101010101		1	001000101001		1	100010100000		3
057	000100101010		2	000101010100		2	010000101001		1
058	000010010001		1	100001010000		3	010000000100		2
059	000010101000		2	001001001001		1	010000001000		2
060	000101010001		1	001000000010		2	010000000101		1

【도 10b】

Data Symbol	MCG1			MCG2			MCG3		
	Code Word MSB	LSB	NCG	Code Word MSB	LSB	NCG	Code Word MSB	LSB	NCG
061	000101001010		2	001000000100		2	100100010000		3
062	000101010010		2	001000000101		1	010001001001		1
063	001000000101		1	100010010000		3	010000101010		2
064	000101010100		2	001001010001		1	010001001010		2
065	000010100001		1	001000001000		2	010001010101		1
066	0001000001010		2	001000101010		2	010001010010		2
067	000100100000		3	001001010101		1	001010101001		1
068	000100001001		1	001001001010		2	001010000100		2
069	000100010010		2	000100010001		1	010001010001		1
070	0001000010100		2	000101000100		2	010001010100		2
071	0001000010101		1	001010001001		1	010010001010		2
072	000100100010		2	001001010010		2	010010010101		1
073	000100100100		2	001001010100		2	010010010010		2
074	000100100101		1	001010010101		1	010000001001		1
075	000100101000		2	0010100001010		2	001010001000		2
076	001000101001		1	000100100001		1	100000010000		3
077	001000000010		2	000101001000		2	010000010001		1
078	000101001000		2	010000010000		3	001010101010		2
079	001001001001		1	000101000001		1	010000001010		2
080	001000000100		2	001000001010		2	001010000101		1
081	001000001000		2	001000010010		2	010010001001		1
082	001001010101		1	001000010101		1	010010010100		2
083	001000001001		1	001010010001		1	010010100010		2
084	001001010001		1	001010010010		2	010010100101		1
085	001000101010		2	001010010100		2	010010100100		2
086	001001001010		2	001010100101		1	010000100001		1
087	001010010101		1	001010100010		2	010000010010		2
088	001001010010		2	001000001001		1	100000100000		3
089	0001000010001		1	001000010100		2	010001000001		1
090	000101000010		2	010000100000		3	010000010100		2
091	001010001001		1	001000010001		1	010000100010		2
092	001001010100		2	001000100010		2	010000010101		1
093	001010001010		2	001000100100		2	100101010000		3
094	001010100101		1	001000100101		1	010010000001		1
095	001010010010		2	010101010000		3	010000100100		2
096	000100100001		1	001000100001		1	010000101000		2
097	0001010000100		2	001000101000		2	010000100101		1
098	001000010000		3	001001000010		2	010001000010		2
099	000101000001		1	001001000101		1	010010010001		1
100	001000001010		2	001001000100		2	010010101000		2
101	001000010010		2	001010100001		1	010010100001		1
102	0001010000101		1	001010100100		2	010100001010		2
103	001010010001		1	001001001000		2	010100010010		2
104	001010010100		2	001010000101		1	010100010101		1
105	001010100010		2	001010000010		2	010100010100		2
106	010000000101		1	010000101001		1	010010101001		1
107	001010100100		2	001010101000		2	010001000100		2
108	001000010001		1	100010100000		3	010100001001		1
109	001000010100		2	010001001001		1	010100000001		1
110	001000100000		3	010000000100		2	010001001000		2
111	001000100001		1	010000001000		2	010010000010		2
112	001000100010		2	010000000101		1	010001000101		1
113	001000100100		2	100100010000		3	010100010001		1
114	001000010101		1	010001010001		1	010100101001		1
115	010000010000		3	010000101010		2	010010000100		2
116	001001000001		1	010001001010		2	010010001000		2
117	001000101000		2	010001010101		1	010010000101		1
118	001001000010		2	010001010010		2	010010101010		2
119	001000100101		1	001001000001		1	010100100001		1
120	001001000100		2	001010000100		2	010100100010		2

【도 10c】

Data Symbol	MCG1			MCG2			MCG3		
	Code Word MSB	LSB	NCG	Code Word MSB	LSB	NCG	Code Word MSB	LSB	NCG
121	001010100001		1	010010001001		1	010101001001		1
122	001010101000		2	010001010100		2	010100000010		2
123	001001001000		2	010010001010		2	010100000100		2
124	001001000101		1	010010010101		1	010100000101		1
125	001010000010		2	010010010010		2	010100001000		2
126	010000101001		1	001010000001		1	010101000001		1
127	010000000100		2	001010001000		2	010100100100		2
128	001010000001		1	100000010000		3	100100100000		3
129	010001001001		1	001010101001		1	100000101001		1
130	010000001000		2	001010101010		2	010100101000		2
131	010000101010		2	010000001010		2	010101000010		2
132	010001010101		1	010000010101		1	010100100101		1
133	0010100001000		2	010010010001		1	010100101010		2
134	010001010001		1	010010010100		2	100001001001		1
135	010001001010		2	010010100010		2	0101010000100		2
136	010001010010		2	010010100101		1	010101010001		1
137	010010010101		1	010010100100		2	100001010001		1
138	010001010100		2	010000001001		1	010101001000		2
139	001010101001		1	010000010010		2	100000001000		2
140	001010000100		2	100000100000		3	010101000101		1
141	0100100001001		1	010000010001		1	0101010001010		2
142	0100100001010		2	010000010100		2	100010001001		1
143	010010010010		2	010000100010		2	100000101010		2
144	010010100101		1	010000100101		1	100001001010		2
145	010010010100		2	010010101000		2	100001010101		1
146	010000001001		1	010000100001		1	100001010010		2
147	001010101010		2	010000100100		2	100000001001		1
148	010000100000		3	010000101000		2	010101010010		2
149	010000010001		1	010001000101		1	100010010001		1
150	010000001010		2	010001000010		2	100001010100		2
151	010000010010		2	010010100001		1	100010001010		2
152	001010000101		1	010100001010		2	100010010101		1
153	010010010001		1	010100001001		1	100010010010		2
154	010010100010		2	0101000010010		2	100000010001		1
155	010010100100		2	0101000010100		2	010101010100		2
156	010100010101		1	010100010101		1	100010010100		2
157	010010101000		2	010100100010		2	100000100001		1
158	010000100001		1	010001000001		1	100000001010		2
159	010000010100		2	010001000100		2	100000010010		2
160	010101010000		3	010100010001		1	010101010101		1
161	010001000001		1	010010000001		1	100010100001		1
162	010000100010		2	010001001000		2	100010100010		2
163	010000100100		2	010010000010		2	100010100100		2
164	010000010101		1	010010000101		1	100010100101		1
165	010100001001		1	010100100100		2	100010101000		2
166	010010000001		1	010010101001		1	100001000001		1
167	010000101000		2	010010000100		2	100000010100		2
168	010001000010		2	010010001000		2	100100001001		1
169	010000100101		1	010100000101		1	100010000001		1
170	0100010000100		2	010010101010		2	100000100010		2
171	010010100001		1	010100100001		1	100000100100		2
172	010100001010		2	010100101000		2	100000010101		1
173	010100010001		1	010100000001		1	100100001010		2
174	010100010010		2	010100000010		2	100010101001		1
175	010100010100		2	010100000100		2	100000101000		2
176	010100100101		1	010101010101		1	100001000010		2
177	010100100010		2	010100001000		2	100000100101		1
178	010010101001		1	010101000001		1	100001000100		2
179	010001001000		2	010101000010		2	100100010001		1
180	010100100100		2	100100100000		3	100100010010		2

【도 10d】

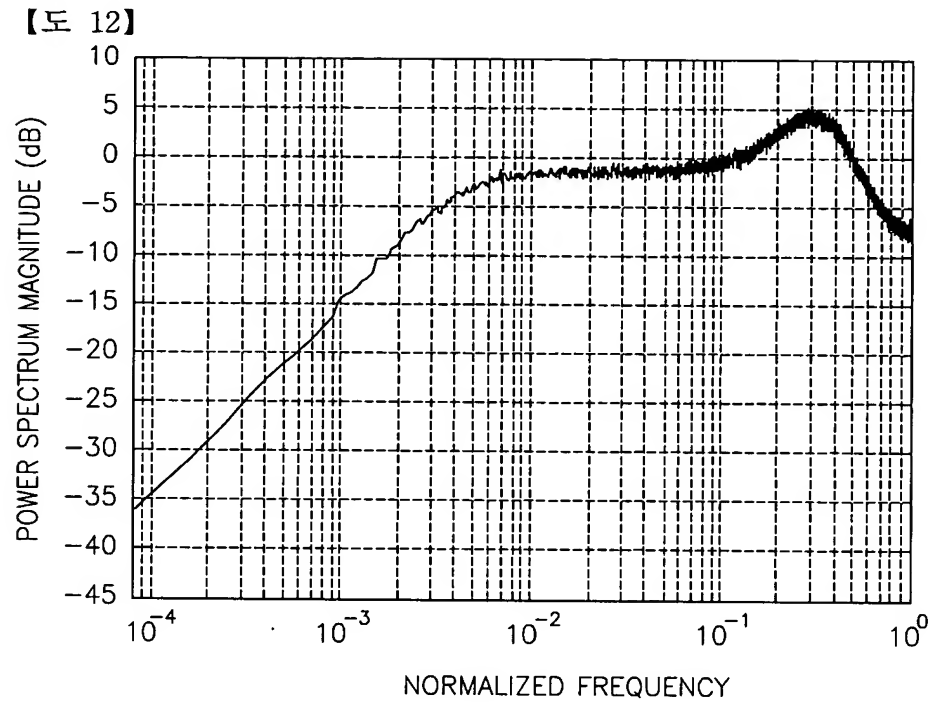
Data Symbol	MCG1			MCG2			MCG3		
	Code Word		NCG	Code Word		NCG	Code Word		NCG
	MSB	LSB		MSB	LSB		MSB	LSB	
181	010100000001		1	100000101001		1	100100100001		1
182	010010000010		2	010101000100		2	100100010100		2
183	010010000100		2	0101010001000		2	1001001000010		2
184	0100010000101		1	010100100101		1	100100010101		1
185	010100101000		2	010100101010		2	100100100100		2
186	010100101001		1	100001001001		1	100100000001		1
187	010010001000		2	100000001000		2	100001001000		2
188	010010101010		2	010100101001		1	100100101000		2
189	010010000101		1	100001010001		1	100100101001		1
190	010100000010		2	100000101010		2	100010000010		2
191	010100100001		1	100001001010		2	100010000100		2
192	010101000010		2	010101000101		1	100001000101		1
193	010101001001		1	010101001001		1	100101001000		2
194	010100000100		2	100010001001		1	100101001001		1
195	010100001000		2	100001010010		2	100010001000		2
196	010100000101		1	100001010100		2	100010101010		2
197	0101000101010		2	100001010101		1	100010000101		1
198	010101000001		1	100010001010		2	100100000010		2
199	010101000100		2	010101010001		1	100101000001		1
200	010101001010		2	010101001010		2	100101000010		2
201	010101010010		2	100010010001		1	100101010001		1
202	010101001000		2	100010010010		2	100100000100		2
203	010101010001		1	100010010100		2	100100001000		2
204	010101000101		1	100010010101		1	100100000101		1
205	101000001010		2	100010100010		2	100100101010		2
206	101000010010		2	100000001001		1	101000000010		2
207	101000010101		1	010101010010		2	100101000100		2
208	101000010100		2	100010101000		2	100101001010		2
209	101000101001		1	100000010001		1	100101010010		2
210	101000000010		2	010101010100		2	101000000100		2
211	101010100000		3	100000001010		2	101000000101		1
212	101001001001		1	100000010101		1	100100100101		1
213	101000000100		2	100010100001		1	101000001010		2
214	101000001000		2	100010100100		2	101000010010		2
215	101000000101		1	100100001010		2	101000010101		1
216	101000001001		1	100010100101		1	101000010100		2
217	101001010001		1	1001000010010		2	101000101001		1
218	101000101010		2	100000100001		1	101000001000		2
219	101001001010		2	100000010010		2	101001010000		3
220	101001010101		1	100100001001		1	101001001001		1
221	101001010010		2	100001000001		1	101000101010		2
222	101000010001		1	100000010100		2	101001001010		2
223	101000100010		2	100000100010		2	101001010101		1
224	101010001001		1	100000100101		1	101010010000		3
225	101001010100		2	100100010100		2	101001010001		1
226	101010001010		2	100010000001		1	101001010010		2
227	101010010101		1	100000100100		2	101001010100		2
228	101010010010		2	100000101000		2	101010010101		1
229	101000100001		1	100001000101		1	101010001010		2
230	101000100100		2	100001000010		2	101000001001		1
231	101000010000		3	100100010001		1	101000100010		2
232	101001000001		1	100100100010		2	101010001001		1
233	101000101000		2	100100100001		1	101010010010		2
234	101001000010		2	100100100100		2	101010010100		2
235	101000100101		1	100100101000		2	101000010001		1
236	101010010001		1	100100010101		1	101000010000		3
237	101010010100		2	100101000010		2	101000100001		1
238	101010100010		2	100010101001		1	101000100100		2
239	101010100101		1	100001000100		2	101000100000		3
240	101010100100		2	100010000010		2	101001000001		1

【도 10e】

Data Symbol	MCG1			MCG2			MCG3		
	Code Word		NCG	Code Word		NCG	Code Word		NCG
	MSB	LSB		MSB	LSB		MSB	LSB	
241	101010000001		1	100100000001		1	101000101000		2
242	101001000100		2	100001001000		2	101001000010		2
243	101000100000		3	100010000100		2	101000100101		1
244	101010101001		1	100010000101		1	101010010001		1
245	101001001000		2	100100000010		2	101001000100		2
246	101010000010		2	100100101001		1	101001001000		2
247	101001000101		1	100010001000		2	101010001000		2
248	101010101000		2	100010101010		2	101010000001		1
249	101010000100		2	100100000101		1	100101010101		1
250	101010001000		2	100100000100		2	101010000010		2
251	101010101010		2	100101000001		1	101010000100		2
252	101010000101		1	100101000100		2	100101000101		1
253	010101010101		1	100100001000		2	100101010100		2
254	101010100001		1	100100101010		2	101001000101		1
255	010101010100		2	100100100101		1	101010000101		1

【도 11】

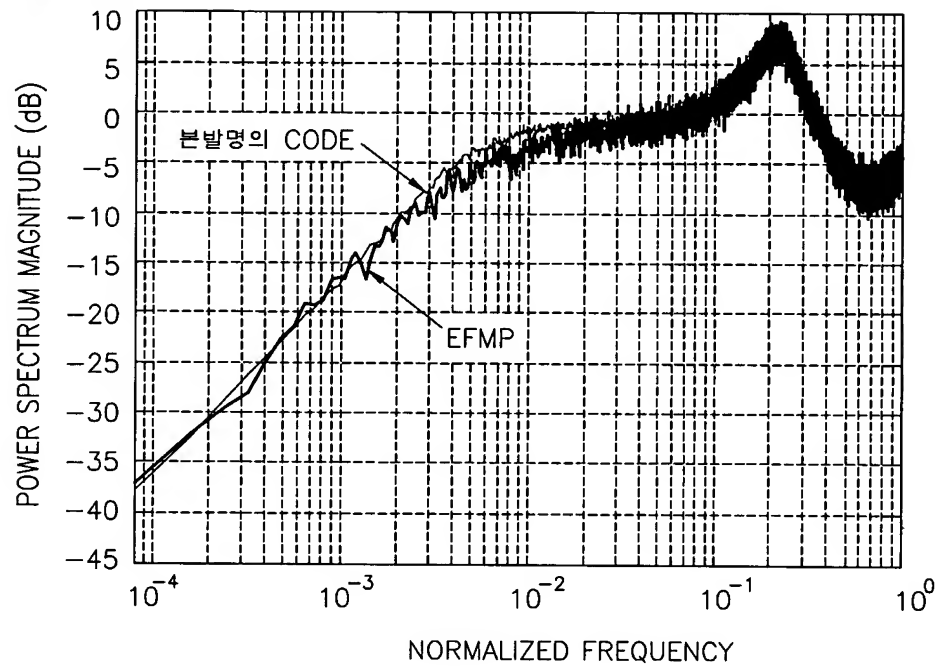
Data Symbol	ACG1			ACG2			ACG3		
	Code Word		NCG	Code Word		NCG	Code Word		NCG
	MSB	LSB		MSB	LSB		MSB	LSB	
000	000001000000		3	000000100000		3	000100000001		1
001	000010000000		3	000000100001		1	000100000010		2
002	000101000000		3	000000100010		2	000100000100		2
003	001001000000		3	000000100100		2	000100000101		1
004	001010000000		3	000000100101		1	000100001000		2
005	010001000000		3	000000101000		2	000100001001		1
006	010010000000		3	000000101001		1	000100001010		2
007	010101000000		3	000000101010		2	000100010000		3
008	101001000000		3	000001000000		3	000100010001		1
009	101010000000		3	000001000001		1	000100010010		2
010	000000010000		3	000001000010		2	000100010100		2
011	000000010001		1	000001000100		2	000100010101		1
012	000000010010		2	000001000101		1	000100100000		3
013	000000010100		2	000001001000		2	000100100001		1
014	000000010101		1	000001001001		1	000100100010		2
015				000001001010		2	000100100100		2
016				000001010000		3	000100100101		1
017				000001010001		1	000100101000		2
018				000001010010		2	000100101001		1
019				000001010100		2	000100101010		2
020				000001010101		1	000101000000		3
021				000010000000		3	000101000001		1
022				000101000000		3	000101000010		2
023				001001000000		3	000101000100		2
024				001010000000		3	000101000101		1
025				010001000000		3	000101001000		2
026				010010000000		3	000101001001		1
027				010101000000		3	000101001010		2
028				100001000000		3	000101010000		3
029				100010000000		3	000101010001		1
030				100101000000		3	000101010010		2
031				100101000101		1	000101010100		2
032				100101001000		2	000101010101		1
033				100101001001		1	001001000000		3
034				100101001010		2	001010000000		3
035				100101010000		3	010001000000		3
036				100101010001		1	010010000000		3
037				100101010010		2	010101000000		3
038				100101010100		2	100001000000		3
039				100101010101		1	100010000000		3
040							100101000000		3
041							101001000000		3
042							101010000000		3
043							101010100000		3
044							101010100001		1
045							101010100010		2
046							101010100100		2
047							101010100101		1
048							101010101000		2
049							101010101001		1
050							101010101010		2



【도 13】

	d	k	$R(m/n)$	$R/C(d,k)$
A-Code	1	7	0.6533	96.2%
B-Code	1	8	0.6531	95.3%
C-Code	1	7	0.6536	96.2%
본 발명의 Code	1	7	0.6650	97.9%

【도 14】



【도 15】

	d	k	R(m/n)	R/C(d,k)
EFMP	2	10	0.5	96.2%
본 발명의 Code	2	10	0.5284	97.9%